

A2

The fare construction process 40 retrieves 44 an interior city from a list of interior cities "C1" that are part of an arbitrary for the airline "A." The fare construction process 40 retrieves 46 a city C2 from the list 38a (via an access to the first hash table 37a) of all of the cities "C2" that form arbitraries C1-C2 for city "C1" for airline "A." The constructed fares process 30 retrieves 48 from the list 38b (via an access to the second hash table 38a) a city "C3" that forms a published fare with city "C2" for airline "A." The fare construction process 40 tries 50 to produce a constructed fare for the C1-C2-C3 city combinations. This embodiment of the fare construction process 40 produces the two-component constructed fares, i.e., one arbitrary combined with one published fare.

The fare construction process 40 will determine 52 if the C1-C2-C3 constructed fare is valid. If it is valid it is stored 54 or otherwise noted, and then a pointer (not shown) is incremented to get the next C3 city 56. If there are more C3 cities 58, the fare construction process will retrieve 48 the next gateway city C3 for which a published fare exists with C2 that is all of the cities that form a published fare with gateway city C2 and the fare construction process 40 will continue. If all cities C3 have been tested, it will then retrieve the next C2 city 60 by incrementing another pointer (not shown) to get the next C2 city. If there are more C2 cities 62, the fare construction process 40 will retrieve 46 the next gateway city C2 for which an arbitrary exists with city C1 and the fare construction process 40 will continue.

If all C2 cities have been tested, it will then retrieve the next C1 city 64 by incrementing another pointer (not shown) to get the next C1 city 44. If there are more C1 cities, the fare construction process will retrieve the next interior city C1 will continue. If all interior cities C1 have been tested, it will then retrieve the next airline 66 by incrementing another pointer (not shown) to get the next airline.

Referring now to FIGS. 4A-4C, a fare construction process 40' to construct two and three-component constructed fares, i.e., two arbitraries combined with one published fare is shown. In the embodiment 40', there is an additional loop. The fare construction process 40' is similar to the process 40 up to when the fare construction process tries 50 to construct a two-component constructed fare. Thus, the fare construction process 40' uses processes 42, 44, 46, 48 as well as 52, 54, 56, 58 and 60, 62, 64, 66 and 68 of FIGS. 3A-3B. The fare construction process 40' retrieves 51a a second interior city C4 from a list of interior cities "C4" that are part

SUB
B2

A2
cont

of an arbitrary that exists for city C3 for the airline "A." The fare construction process 40 retrieves 51a a city C4 from the list 38c (via an access to the third hash table 37c) of all of the cities "C4" that form arbitraries C3-C4 for gateway city "C3" for airline "A." The fare construction process 40' tries 51b to produce a constructed fare for the C1-C2-C3-C4 city combinations. The fare construction process 40 will determine 51c if the C1-C2-C3-C4 constructed fare is valid. If it is valid it is stored 54a or otherwise noted, and then a pointer (not shown) is incremented to get 51d the next C4 city. If there are more C4 interior cities 51e, the fare construction process 40' will retrieve the next interior city C4 and the fare construction process will continue. If all cities C4 have been tested, it will continue from testing 52 the C1-C2-C3 two component constructed fare as described above. This embodiment 40' produces two-component and three-component constructed fares, i.e., two arbitraries combined with one published fare.

Replace the paragraphs beginning at page 14, line 15, and ending on page 15, line 8 with the following rewritten paragraphs:

A3

Referring to FIG. 5, in determining whether or not a fare can combine with an arbitrary 94, the fare construction process 40 checks the "PO2" records corresponding to the fare 96. If the PO2 records do not list the arbitrary's fare code, the arbitrary cannot combine with the fare, and the construction is prohibited. Otherwise, the fare construction process 40 returns a pass 98. The conventional process of finding the set of PO2 records matching a particular fare is time consuming. Alternatively, the check of fare construction tables can be done on a batch basis after all potential constructed fare combinations have been determined.

To save time when constructing large numbers of fares, a PO2 procedure retrieval 80 uses a process called memoization. Memoization is a technique for speeding up certain kinds of algorithms. If an expensive procedure is called many times, and if the procedure's output depends only on the input (i.e., the answer is not dependent on any external factors, such as the current time), then memoization can be used. To memoize PO2 retrieval, the PO2 procedure call 80 forms a query 82 involving the constructed fare. The memoization retrieval process 80 has a store 88 of past queries and associated answers. If the memoization retrieval process 80 determines 84 that the query has been stored, it is retrieved from the store 92. Otherwise, a procedure call 86 for the PO2 record is produced and used to access 90 the record from a remote